

Computational Techniques for Reachability Analysis of Partially Observable Discrete Time Stochastic Hybrid Systems

Kendra Lesser, *Student Member, IEEE*, Meeko Oishi, *Member, IEEE*

Abstract

Reachability analysis of hybrid systems has been used as a safety verification tool to assess offline whether the state of a system is capable of remaining within a designated safe region for a given time horizon. Although it has been applied to stochastic hybrid systems, little work has been done on the equally important problem of reachability under incomplete or noisy measurements of the state. Further, there are currently no computational methods or results for reachability analysis of partially observable discrete time stochastic hybrid systems. We provide the first numerical results for solving this problem, by drawing upon existing literature on continuous state partially observable Markov decision processes (POMDPs). We first prove that the value function for the reachability problem (with a multiplicative cost structure) is piecewise-linear and convex, just as for discrete state POMDPs with an additive cost function. Because of these properties, we are able to extend existing point-based value iteration techniques to the reachability problem, demonstrating its applicability on a benchmark temperature regulation problem.

Index Terms

Markov decision processes, optimal control, partial observability, reachability, stochastic hybrid systems, value function

K. Lesser and M. Oishi are with the Department of Electrical and Computer Engineering, University of New Mexico, Albuquerque, NM 87131 USA; e-mail: {lesser, oishi}@unm.edu; Tel.: +1 505 353 2424; Fax: +1 505 277 0299.

This research was funded by National Science Foundation (NSF) Career Award CMMI-1254990, NSF Award CPS-1329878, NSA Science of Security Lablet at North Carolina State University (subaward to the University of New Mexico), and start-up funding from the Department of Electrical and Computer Engineering, University of New Mexico

I. INTRODUCTION

Stochastic hybrid systems provide a modeling framework well-suited for a wide range of applications. They allow for versatile dynamics that incorporate codependent discrete and continuous states, often exhibited in systems that may switch between different modes of operation, and account for probabilistic uncertainty in those dynamics. Having such a flexible framework is particularly important in the context of safety verification, where the assessment of a system's ability to meet rigorous safety requirements must be as accurate as possible. Indeed, reachability analysis (determining whether a system's state stays within a given safe region and/or reaches a desired target set within some finite time horizon) for hybrid systems has been studied extensively [1], [2], [3], [4], [5].

Equally important to safety verification, however, is the consideration of not only stochastic and complex dynamics, but also of noisy or incomplete measurements of the state. While there has been some work on deterministic hybrid systems with incomplete information [6] or uncertain hybrid systems with the assumption of a worst-case disturbance [7], reachability analysis of a partially observable stochastic hybrid system has been approached only recently [8], [9], and only theoretically; there are currently no computational results for reachability analysis of *partially* observable stochastic hybrid systems.

Computational results for reachability analysis of perfectly observable stochastic hybrid systems are also limited. The reachability problem for discrete time stochastic hybrid systems (DTSHS) is a multiplicative cost stochastic optimal control problem [4], which can equivalently be formulated as a Markov decision process (MDP). Solutions via dynamic programming produce a state-based feedback controller designed to optimize the system according to some cost function (see [10]). Unfortunately, dynamic programming requires evaluation of the value function over all possible states, which is infinite when those states are continuous. Discretization procedures can be employed to impose a finite number of states, as in [11], which presents a formal adaptive gridding procedure for verification of DTSHS. Gridding methods are unfortunately subject to the “curse of dimensionality” and can lead to an unacceptable number of states that render the dynamic program impossible to implement. Other approximate solution strategies include approximate dynamic programming, where the value function of the dynamic program is approximated by a set of basis functions, as in [12]. Even so, current applications are limited

to those with only a few discrete and continuous states.

The reachability problem for a partially observable DTSHS (PODTSHS) can similarly be formulated as a partially observable MDP (POMDP). However, POMDPs are plagued by dimensionality on an even greater scale than MDPs. The common approach to solving POMDPs is to replace the growing history of observations and actions by a sufficient statistic, often called the belief state, which, for a POMDP with an additive cost function, is the distribution of the current state conditioned on all past observations and actions [10]. This belief state is treated as the perfectly observed true state, and MDP solution methods can then be applied. However, given a continuous state space, the belief state is now a continuous function defined over an infinite domain, and it is impossible to enumerate over all such functions. Therefore the study of efficient, approximate solutions to POMDPs is essential.

Although finding the solution to a general POMDP is hard [13], many algorithms for approximating solutions to finite state POMDPs have been developed. These mainly rely on point-based value iteration (PBVI) schemes that only consider a subset of the belief space to update the value function (for a survey of PBVI algorithms, see [14]). Such methods must be tailored to continuous state POMDPs because of the dimensionality of the belief state.

Many existing methods for continuous state POMDPs assume the belief state is Gaussian, such as in [15], [16], and represent the belief state in a parameterized form which is then discretized and solved as a discrete state MDP. For problems where the belief cannot be represented adequately as a single Gaussian, however, these techniques are subject to the same curse of dimensionality as large discrete state MDPs. Other methods use a Gaussian representation of the belief state to find *locally* optimal solutions, either by parameterizing the value function [17] or by assuming maximum-likelihood observations [18] [19]. An extension of [18] to non-Gaussian beliefs was presented in [20], where the belief states are estimated using sampling. Another sampling-based method that allows for a non-Gaussian belief state is given by [21], where the belief state is updated according to a particle filter, and Monte Carlo methods and nearest-neighbor approximations estimate the value function.

PBVI techniques have also been extended to the case of continuous states in [22], which showed that for continuous states and discrete actions and observations, the value function remains piecewise-linear and convex (as was shown for discrete state POMDPs by [23]). These properties can be exploited to approximate the value function by a finite set of “ α -functions,”

which are a function of the true state of the system, and represent the value of being in that state, including the future expected rewards assuming optimal actions are taken. Further, by representing these α -functions and the belief states as linear combinations of Gaussians, updating the belief state and value function can be done in closed form. This technique was extended to hybrid domains, where the discrete mode is hidden and the belief state is a function only of the continuous variable [24]. The authors of [22] also showed that the belief state can be approximated using a particle filter rather than as a sum of Gaussians, and the continuous state PBVI method still applied.

The reachability problem for PODTSHS further complicates the already difficult problem of solving continuous state POMDPs. As was shown in both [8] and [9], the belief state of the PODTSHS is no longer just the conditional distribution of the current state of the system, but must also include the distribution of a binary variable indicating whether the state of the system has remained within a safe region up to the previous time step. This, coupled with the stochastic hybrid system dynamics, makes representing the belief state as a single Gaussian impossible, and using sampling to update the belief can be expensive.

Therefore, as the first investigation into approximate solutions to the reachability problem for PODTSHS, we consider continuous state PBVI techniques as in [22] and [24]. These techniques are amenable to stochastic hybrid dynamics, and have already been demonstrated as effective in hybrid domains with a hidden discrete state. In this paper we present several contributions to the solution of safety verification problems for PODTSHS. First, we show that even with the multiplicative cost structure of the reachability problem, as in [4] and [9], the value function is piecewise-linear and convex under the assumption of discrete actions and observations. Further, the belief state, defined over a hybrid domain, and value function maintain the closedness property of the belief and value function updates, when they are represented as weighted sums of Gaussians. Proving the preservation of these “nice” properties enables the application of existing POMDP solution techniques. Second, we exploit the structure of the belief state and value function to extend the technique of [22] and [24] to the reachability problem. We outline a solution method, and demonstrate its effectiveness on a temperature regulation problem.

The rest of the paper is organized as follows. Section II-A defines a PODTSHS, and formulates the reachability problem. Sections II-B and II-C provide an overview of POMDPs and their exact solution, and point-based value iteration techniques, respectively. PODTSHSs and POMDPs are

related in Section II-D. Section III establishes properties of the value function, demonstrates how PBVI techniques can be used to solve the reachability problem for PODTSHS, and also provides a bound on the error introduced in approximating the true value function. Section III also shows that the value function and belief updates preserve the Gaussian representation. Section IV provides numerical results using a benchmark temperature regulation problem, and discusses computational issues. Section V provides concluding remarks and future directions.

II. BACKGROUND

A. Reachability for PODTSHS

A hybrid system is characterized by a set of both discrete and continuous states with interacting dynamics: the discrete state may affect the evolution of the continuous dynamics, and the continuous dynamics may affect when the discrete state changes. In the case of a DTSHS, both the discrete and continuous dynamics may be characterized by stochastic kernels, the product of which determines the stochastic transition kernel governing the combined discrete/continuous state of the system. We present a slightly modified definition of a DTSHS first introduced in [4].

Definition 1. (*Discrete Time Stochastic Hybrid System \mathcal{H}*). A DTSHS is a tuple $\mathcal{H} = (\mathcal{X}, \mathcal{Q}, \mathcal{U}, T_x, T_q)$ where

- 1) $\mathcal{X} \subseteq \mathbb{R}^n$ is a set of continuous states
- 2) $\mathcal{Q} = \{q_1, q_2, \dots, q_{N_q}\}$ is a finite set of discrete states with cardinality N_q , with $\mathcal{S} = \mathcal{X} \times \mathcal{Q}$ the hybrid state space
- 3) \mathcal{U} is a compact Borel space which contains all possible control inputs affecting discrete and continuous state transitions
- 4) $T_x : \mathcal{B}(\mathbb{R}^n) \times \mathcal{Q} \times \mathcal{S} \times \mathcal{U} \rightarrow [0, 1]$ is a Borel-measurable stochastic kernel which assigns a probability measure to x_{t+1} given $s_k = (x_t, q_t), u_t, q_{t+1} \forall t: T_x(dx_{t+1} \in B \mid q_{t+1}, s_t, u_t)$ where $B \in \mathcal{B}(\mathbb{R}^n)$, the Borel σ -algebra on \mathbb{R}^n
- 5) $T_q : \mathcal{Q} \times \mathcal{S} \times \mathcal{U} \rightarrow [0, 1]$ is a discrete transition kernel assigning a probability distribution to q_{t+1} given $x_t, q_t, u_t, \forall t$

Kernels T_x and T_q can be combined for ease of notation to produce one hybrid state transition

kernel, denoted $\tau(\cdot)$, given by:

$$\tau(ds' | s, u) = T_x(dx' | x, q, u, q')T_q(q' | x, q, u) \quad (1)$$

The discrete state q_{t+1} update depends on q_t , x_t and u_t , and the continuous state x_{t+1} update depends on x_t , u_t , and according to the specific problem may also be governed by q_t , q_{t+1} , or both. For ease of notation we assume that the discrete state updates first, and the updated discrete state affects the continuous state, i.e. that $T_x(dx_{t+1} | x_t, u_t, q_{t+1})$, although modifying T_x to include q_t would not alter any subsequent results.

For a PODTSHS, it is assumed that only an observation process is available to the controller, of the form $y_t = (y_t^x, y_t^q)$, where y_t^x is associated with x_t , and y_t^q with q_t . While y_t^x could be continuous, for computational purposes we assume that it is discrete-valued, even though x_t is continuous (which could arise simply by discretizing the observation process). The observation process is given by

$$y_t^x = h(x_t, u_{t-1}) + v_t \quad (2)$$

$$y_t^q \sim Q_{q, y^q}(u) \quad (3)$$

The probability that $y_t^q = n$, $P[y_t^q = n | q_t = q, u_{t-1} = u] = Q_{q, n}(u)$, is given by the state transition matrix $Q(u)$ which is dependent on the control input u . For the continuous state observation y_t^x that is continuous-valued, it is subject to additive noise v_t , which is independent and identically distributed with positive density $\varphi(v)$ (i.e. Gaussian), and the function h is assumed to be bounded and continuous. Otherwise we assume y_t^x has a state transition matrix similar to Q , and we will write $\varphi(y^x | x, u)$ to express the conditional discrete distribution of y^x . The filtrations \mathcal{G}_t and \mathcal{Y}_t are generated by the sequences $\{s_0, \dots, s_t, y_1, \dots, y_{t-1}\}$ and $\{y_1, \dots, y_t\}$, respectively. We also assume an initial Borel-measurable density on $s_0 = (x_0, q_0)$, $s_0 \sim \rho(x, q) \in P(\mathcal{S})$, i.e. that ρ lies in the space of all probability measures on \mathcal{S} . Finally, based on ρ , τ , φ , and $Q(u)$, the probability measure \mathbb{P}^π is induced by the control policy π defined over the full state space Ω , which includes s_t and y_t for all t .

Next, we present a cost function to analyze the reachability of the partially observable DTSHS, i.e. the ability of the state to remain within some safe or desired region of the state space. We want to find both a control policy that maximizes the probability of the state remaining within that desired set, as well as an estimate of that probability. As in [4], this problem can be formulated

as a stochastic optimal control problem. For a Borel set $K \subseteq \mathcal{X} \times \mathcal{Q}$, terminal time T , and predefined policy π , define the cost function as

$$r_K(\pi) = \mathbb{P}^\pi[s_t \in K \forall t = 0, \dots, T] \quad (4)$$

Since for a random variable X , $\mathbb{P}[x \in A] = \mathbb{E}[\mathbf{1}_A(x)]$, with \mathbb{E} denoting expected value and indicator function $\mathbf{1}_A(x) = 1$ if $x \in A$ and $\mathbf{1}_A(x) = 0$ otherwise, (4) is rewritten as in [4]:

$$r_K(\pi) = \mathbb{E}^\pi \left[\prod_{t=0}^T \mathbf{1}_K(s_t) \right] \quad (5)$$

The expected value is taken with respect to the measure \mathbb{P}^π , hence the notation \mathbb{E}^π . We want to maximize $r_K(\pi)$ with respect to the control policy π . The set Π of admissible policies will be restricted to non-randomized policies, i.e. in which $\pi(y_t)$ generates one control input u_t with probability 1. The optimal policy π^* is then given by

$$\pi^* = \arg \sup_{\pi \in \Pi} \{r_K(\pi)\} \quad (6)$$

We can now formally define the problem we wish to solve.

Problem 1. Consider a DTSHS \mathcal{H} (defined in Definition 1) with observations (2) - (3) and initial distribution $\rho(x, q) \in P(\mathcal{S})$. Given a safe set K and time horizon T we would like to

- 1) Compute the maximal probability of remaining within K for T time steps, given by $\sup_{\pi} r_K(\pi)$.
- 2) Compute the optimal policy π^* such that $\sup_{\pi} r_K(\pi) = r_K(\pi^*)$.

If the maximal probability and optimal policy cannot be computed exactly (which is quite likely [13]), an approximation producing a suboptimal policy and lower bound on the maximal reachability probability are desired.

B. Optimal Control of POMDPs

POMDPs provide a framework for analyzing a discrete time system whose state depends on the actions of an agent (controller), who is trying to drive the state to optimize some objective. The state evolves stochastically and is Markovian (the state at the next time step depends only on the current state and action). Further, in choosing actions, the agent can not directly observe the state of the system, instead only having access to an observation process. We first define a POMDP with discrete states, actions, and observations, and an additive cost function. The theory

and solution techniques for this type of POMDP provide the foundation for our extension to a PODTSHS and the solution of Problem 1.

Definition 2. (POMDP \mathcal{G}) A POMDP is a tuple $\mathcal{G} = (\mathcal{S}, \mathcal{U}, \mathcal{Y}, \tau, \psi, R)$ where

- 1) \mathcal{S} is a set of discrete states
- 2) \mathcal{U} is a discrete set of possible actions the agent can take
- 3) \mathcal{Y} is a set of discrete observations
- 4) $\tau : \mathcal{S} \times \mathcal{S} \times \mathcal{U} \rightarrow [0, 1]$ is a state transition function assigning a probability distribution to state s_{t+1} given state s_t and action u_t for all t , $\tau(s_{t+1} \mid s_t, u_t)$
- 5) $\psi : \mathcal{Y} \times \mathcal{S} \times \mathcal{U} \rightarrow [0, 1]$ is an observation function assigning a probability distribution to observation y_t given state s_t and action u_t for all t , $\psi(y_t \mid s_t, u_t)$
- 6) $R : \mathcal{S} \times \mathcal{U} \rightarrow \mathbb{R}$ is a function assigning a reward (which we define as being in the set of all real numbers \mathbb{R} , although this could be generalized to any space) at each time step t , given the current state s_t and action u_t , $R(s_t, u_t)$

The goal for the POMDP \mathcal{G} is to maximize the expected sum of rewards over a (possibly infinite) time horizon T by optimally choosing a sequence of control actions $\bar{u} = \{u_1, u_2, \dots\}$.

$$\max_{\bar{u}} \mathbb{E} \left[\sum_{t=0}^T R(s_t, u_t) \right] \quad (7)$$

Rather than keeping track of all past observations and actions in order to make an optimal decision at time t , a belief state is used instead, which summarizes all available information up to time t . The belief state is a *sufficient statistic* for the set of all observations and actions $\{u_1, \dots, u_{t-1}, y_1, \dots, y_t\}$ because it condenses all information necessary for making optimal decisions [10]. In the case of an additive cost POMDP, the belief state is a probability density function that describes the probability of being in state s given all past observations and actions, $b(s_t) = P[s_t \mid u_1, \dots, u_{t-1}, y_1, \dots, y_t]$. Treating the belief state as the true state of the system, \mathcal{G} can be equivalently solved as a perfect state information MDP. An optimal policy π^* for the POMDP is defined in terms of the belief state, and maps beliefs to actions: $\pi^* : \mathcal{B} \rightarrow \mathcal{U}$.

The optimal policy can be found by using a value function over the space of beliefs \mathcal{B} , which describes the cumulative reward from time t to the final time T (or over $T - t + 1$ time steps), for a particular belief state b , and assuming the system behaves optimally from time $t + 1$ to T . The control u_t is chosen to maximize the value function at a specific belief b . Because the

value function assumes only optimal actions are taken starting at time $t + 1$, it can be defined recursively using the optimal value function at time $t + 1$.

$$V_t^*(b) = \max_{u \in \mathcal{U}} \left\{ \sum_s R(s, u) b(s) + \sum_y V_{t+1}^*(M_{y,u}[b]) P[y | u, b] \right\} \quad (8)$$

The transition operator $M_{y,u}[b]$ provides the next belief state b_{t+1} given the current observation, action, and belief state. Sondik [23] first showed that for a finite horizon $T < \infty$, the value functions are piecewise-linear and convex, and thus can be expressed as

$$V_t^*(b) = \max_{\alpha_t^i \in \Gamma_t} \sum_s \alpha_t^i(s) b(s) \quad (9)$$

The functions $\alpha_t^i \in \Gamma_t$, or “ α -vectors”, can be thought of as representing a policy tree starting from a specific action u and state s , which then specifies optimal actions conditioned on observations for the following time steps $t + 1$ to T . The α -vectors thus characterize the current value of being in state s and taking action u , plus the expected sum of future rewards assuming all subsequent actions are chosen optimally. Because each α -vector is associated with a specific action, by picking the α -vector that maximizes $\sum_s \alpha_t^i(s) b(s)$, we are also defining the optimal policy for belief b at time t .

In order to calculate the value function and optimal policy for all times t , all that is required are the complete sets of α -vectors, Γ_t , for all t . Unfortunately, the number of α -vectors grows exponentially with t . The α -vectors at time t are computed recursively from the α -vectors calculated at time $t + 1$. For each action, we observe one of $|\mathcal{Y}|$ observations (where $|\cdot|$ indicates the cardinality of the set), and for each of those observations there is a subsequent α -vector defined at time $t + 1$, resulting in $|\mathcal{U}| |\Gamma_{t+1}|^{|\mathcal{Y}|}$ α -vectors at time t .

Often, some of the α -vectors are completely *dominated* by another α -vector or set of α -vectors (where $\sum_s \alpha_t^j(s) b(s) < \sum_s \alpha_t^k(s) b(s)$ for all $b(s)$ implies α_t^j is dominated by α_t^k). While those dominated vectors clearly do not need to be included in the set Γ_t , finding the unnecessary α -vectors is also computationally expensive. A number of approximate solution techniques, including point-based value iteration, have been developed.

C. Point-Based Value Iteration

Point-Based Value Iteration (PBVI) computes the value function only over a finite subset $B \subset \mathcal{B}$. The general idea is to generate a collection of points $b \in \mathcal{B}$, and for each of these points

perform a “backup” operation to get a new estimate of the value function at that point. Most PBVI approaches use the same method of updating the value function at each belief point (the “backup” operation) and are distinguished by how they select the subset B (see [14]). Here we outline the method of estimating the value function presuming a set B has already been selected. A discussion of various methods for selecting B can be found in [25] and [14].

One α -vector must be generated for each belief point $b^i \in B$, $B = (b^0, b^1, \dots, b^n)$, so that $\tilde{\Gamma}_t = (\alpha_t^0, \alpha_t^1, \dots, \alpha_t^n)$ for all t . We assume that an α -vector α_t^j corresponding to b^j will apply to all belief points in a region around b^j (i.e. for any b in a neighborhood of b^j the same action will likely be optimal). Hence the value at some b not necessarily in B can be approximated by

$$V_t^*(b) \approx \max_{\alpha_t^i \in \tilde{\Gamma}_t} \sum_s \alpha_t^i(s) b(s)$$

as in (9) but with a restricted set $\tilde{\Gamma}_t \subset \Gamma_t$. The set $\tilde{\Gamma}_t$ is generated recursively from $\tilde{\Gamma}_{t+1}$, but without enumeration over all possible combinations of observations and subsequent α -vectors in $\tilde{\Gamma}_{t+1}$ (the full policy tree starting at time t).

For a specific $b \in B$, the value function at time t can be approximated as follows (see, e.g., [14] for more detail):

$$V_t^*(b) = \max_{u \in \mathcal{U}} \left\{ \sum_{s \in \mathcal{S}} R(s, u) b(s) + \sum_y V_{t+1}^*(M_{y,u}[b]) P[y | u, b] \right\} \quad (10)$$

$$= \max_{u \in \mathcal{U}} \left\{ \sum_{s \in \mathcal{S}} R(s, u) b(s) + \sum_y \max_{\alpha_t^i \in \tilde{\Gamma}_t} \sum_{s' \in \mathcal{S}} \alpha_{t+1}^i(s') M_{y,u}[b](s') P[y | u, b] \right\} \quad (11)$$

$$= \max_{u \in \mathcal{U}} \left\{ \sum_{s \in \mathcal{S}} R(s, u) b(s) + \sum_y \max_{\alpha_{t+1}^i \in \tilde{\Gamma}_{t+1}} \sum_{s' \in \mathcal{S}} \alpha_{t+1}^i(s') \psi(y | s', u) \sum_s \tau(s' | s, u) b(s) \right\} \quad (12)$$

where (12) follows from expanding the operator M . We define the restricted set $\tilde{\Gamma}_t$ using the following expressions

$$\alpha_{y,u}^i(s) = \sum_{s' \in \mathcal{S}} \alpha_{t+1}^i(s') \psi(y | s', u) \tau(s' | s, u) \quad (13)$$

$$\alpha_{y,u,b}(s) = \arg \max_i \sum_s \alpha_{y,u}^i(s) b(s) \quad (14)$$

to obtain

$$\tilde{\Gamma}_t = \bigcup_{b \in B} \left\{ R(s, u) + \sum_y \alpha_{y,u,b}(s) \right\}_{\forall u \in \mathcal{U}} \quad (15)$$

The function (13) is an α -function corresponding to a specific action u and observation y (representing the value of being in state s given y is observed and action u is taken). For a given belief state b , (14) is the optimal function $\alpha_{y,u}^i$ for that belief state given y is observed and action u is taken. Summing over all observations \mathcal{Y} (essentially taking the expected value with respect to y) and then taking the union over all belief states in B and actions $u \in \mathcal{U}$ produces (15), the set of α -functions at time t .

Finally, we define the backup operator for a specific belief point b using the set $\tilde{\Gamma}_t$ (15) as

$$\text{backup}(b) = \arg \max_{\alpha_t^i \in \tilde{\Gamma}_t} \sum_{s \in \mathcal{S}} \alpha_t^i(s) b(s) \quad (16)$$

Note that we can now define the optimal value function (12) as

$$V_t^*(b) = \sum_s (\text{backup}(b)(s) \times b(s)) \quad (17)$$

The overall PBVI algorithm then consists of selecting a set of belief points B , and repeatedly applying (16) to each element of B . In the case of a finite horizon of length T , the backup operator will be applied T times, and for an infinite horizon, the backup operator will be applied until some tolerance level is reached (for example, where $\|V_{n+1}(b) - V_n(b)\| < \epsilon$).

The above derivations apply to a model with discrete state, action, and observation spaces, but [22] actually shows that the same technique applies to a POMDP with a continuous state space and discrete observation and action spaces. In this case, the α -vectors are replaced by α -functions defined over the continuous space \mathcal{S} . Because the observations and actions are assumed discrete, there are a finite number of these α -functions, and so the value function is still piecewise-linear and convex, but now with respect to the α -functions. In this case, the optimal value function may instead be represented as $V_t^*(s) = \sup_{\alpha_t^i \in \tilde{\Gamma}_t} \int_{\mathcal{S}} \alpha_t^i(s) b(s) ds$.

When replacing \mathcal{S} with a continuous state space, all of the above derivations hold, but all summations over \mathcal{S} are replaced by integrals. To generalize from the purely discrete case, [22] uses inner product notation rather than a summation or integral, so that (16) would instead be written as

$$\text{backup}(b) = \arg \max_{\alpha_t^i \in \tilde{\Gamma}_t} \langle \alpha_t^i, b \rangle$$

We maintain this notation in our derivations, where in the case of a hybrid state space with continuous state x and discrete state q , $\langle f, g \rangle = \sum_q \int f(x, q) g(x, q) dx$ for well-defined functions f and g .

All of our derivations will assume discrete actions, and discrete (or discretized) observations. If the assumption of discrete actions and observations is dropped, the value function is still convex but is no longer piecewise-linear (since there are an infinite number of α -functions at any given time step). The authors of [22] show, however, that a PBVI algorithm can still be applied to estimate the value functions by carefully sampling from the observation and action spaces. While our method can also be extended to continuous actions and observations, we assume they are discrete for clarity and completeness of subsequent derivations.

D. Relating Problem 1 to a POMDP

We write the PODTSHS of Problem 1 as a POMDP, which we denote $\mathcal{G} - \text{hybrid}$, with hybrid state space $\mathcal{S} = \mathcal{X} \times \mathcal{Q}$, control space \mathcal{U} , hybrid observation space $\mathcal{Y} = \mathcal{Y}^x \times \mathcal{Y}^q$, state transition function τ given by (1), and observation model $\psi(y \mid s, u) = Q_{q, y^q}(u) \varphi(y^x - h(x, u))$. The reward function is given by $R(s_t, u_t) = \mathbf{1}_K(s_t)$. Note, however, that in contrast to the maximization over a sum of $R(s_t, u_t)$ as in (7) for POMDP \mathcal{G} , we want to maximize the product for $\mathcal{G} - \text{hybrid}$, as described in Problem 1.

We then reformulate $\mathcal{G} - \text{hybrid}$ into an equivalent perfect state information MDP, in the same fashion as for POMDP \mathcal{G} , by redefining the state of the system in terms of a sufficient statistic, or belief state. However, because the cost function (5) is multiplicative rather than additive, the posterior distribution of the state at time t given all available information up to time t is no longer valid. In [9], we developed an appropriate sufficient statistic to solve (5) as a perfect state information problem using standard dynamic programming techniques.

In summary, a change of measure, \mathbb{P}^\dagger , makes the observation processes $\{y_t^x\}$ and $\{y_t^q\}$ each identically distributed and independent of $\{x_t\}$ and $\{q_t\}$, respectively, via the Radon-Nikodym derivative [9] [26], such that

$$\left. \frac{d\mathbb{P}^\pi}{d\mathbb{P}^\dagger} \right|_{\mathcal{G}_t} = \Lambda_t \quad (18)$$

where

$$\Lambda_t = \prod_{l=1}^t \frac{\varphi(y_l^x - h(x_l, u_{l-1})) Q_{q_l, y_l^q}(u_{l-1})}{\varphi(y_l^x)^{\frac{1}{N_q}}}$$

The change of measure facilitates sampling to generate the belief states. The sufficient statistic,

$\sigma(x, q)$, can be defined as

$$\sigma_t(x, q) = \mathbb{E}^\dagger \left[\mathbf{1}_q(q_t) \mathbf{1}_x(x_t) \prod_{i=0}^{t-1} \mathbf{1}_K(s_i) \Lambda_t \middle| \mathcal{Y}_t \right], \quad (19)$$

a modification of the posterior distribution, that represents an unnormalized conditional density of the current state joined with the probability that all previous states are in K . The sufficient statistic can be updated recursively using a bounded linear operator Φ :

$$\begin{cases} \sigma_0(x, q) = \rho(x, q) \\ \sigma_t(x, q) = \Phi_{y,u}[\sigma_{t-1}](x, q) \end{cases} \quad (20)$$

where $\Phi_{y,u}[\sigma]$ is given by

$$\Phi_{y,u}[\sigma](x', q') = \sum_{q \in \mathcal{Q}} N_{y^q} N_{y^x} Q_{q', y^q}(u) \int_{\mathbb{R}^n} \mathbf{1}_K(x, q) \varphi(y^x | x', u) \tau(x', q' | x, q, u) \sigma(x, q) dx \quad (21)$$

in the case of discrete observations y^x , with N_{y^q} the number of possible observations of discrete mode q , and N_{y^x} the number of possible observations of continuous state x .

The dynamic programming recursion to solve for (5) and (6),

$$\begin{cases} V_T^*(\sigma) = \langle \sigma, \mathbf{1}_K \rangle \\ V_t^*(\sigma) = \sup_{u \in \mathcal{U}} \mathbb{E}^\dagger [V_{t+1}^*(\Phi_{y,u}[\sigma])] \end{cases} \quad (22)$$

first evaluates the value function $V_T^*(\sigma)$ in terms of the sufficient statistic σ , then recursively solves $V_{T-1}^*(\sigma)$, $V_{T-2}^*(\sigma)$, etc., ultimately resulting in $V_0^*(\rho) = \sup_{\pi \in \Pi} r_K(\pi)$ (see [9] for proof that this is true). We note that [8], [27] showed that the reachability problem can be equivalently formulated as an additive cost optimization by modifying the state of the system to include a binary variable indicating whether the state has remained within the safe region up to the previous time. The authors of [8] developed and then used this additive cost formulation to generate the sufficient statistic for a partially observable DTSHS as the posterior distribution of the modified state. In [9], we showed its equivalence to the multiplicative cost formulation and sufficient statistic.

We write the recursive relationship between the value functions using operator notation,

$$V_t^* = H[V_{t+1}^*] \quad (23)$$

with $H[V] = \sup_{u \in \mathcal{U}} \mathbb{E}^\dagger [V(\Phi_{y,u}[\sigma])]$ as in (22). A useful property of H , which we will use later, is that it is a nonexpansion, meaning

$$\|H[V] - H[U]\|_\infty \leq \|V - U\|_\infty \quad (24)$$

The proof of (24) is straightforward, and hence omitted.

Similarly to the POMDP \mathcal{G} , the value function in (22) for $\mathcal{G} - \text{hybrid}$ must be solved for all functions σ , which lie in an infinite dimensional space. This clearly cannot be solved directly. However, we will show that $\mathcal{G} - \text{hybrid}$ maintains the properties of the POMDP \mathcal{G} , i.e. that the value function is piecewise-linear and convex, and can be expressed as in (9), but with a hybrid state s . In turn, we can use PBVI techniques to approximate the solution to Problem 1.

III. POINT-BASED VALUE ITERATION FOR HYBRID DYNAMICS AND MULTIPLICATIVE COST

A. Properties of the Value Function

We first demonstrate that the value function for Problem 1 is convex for a hybrid state space with possibly continuous (or hybrid) actions and observations, and that the value function is also piecewise-linear in the case of purely discrete actions and observations.

Lemma 1. *The value function (22) is convex in σ for all k .*

Proof: By induction, at time T for $0 \leq \lambda \leq 1$

$$\begin{aligned} V_T^*(\lambda\sigma_1 + (1-\lambda)\sigma_2) &= \sum_{q \in \mathcal{Q}} \int_{\mathbb{R}^n} \mathbf{1}_K(x, q) [\lambda\sigma_1(x, q) + (1-\lambda)\sigma_2(x, q)] dx \\ &= \lambda V_T^*(\sigma_1) + (1-\lambda)V_T^*(\sigma_2) \end{aligned}$$

Assuming $V_{t+1}^*(\sigma)$ is convex in σ

$$\begin{aligned} V_t^*(\lambda\sigma_1 + (1-\lambda)\sigma_2) &= \sup_{u \in \mathcal{U}} \sum_{y^q} \int_{\mathbb{R}^n} V_{t+1}^*(\Phi_{y,u}[\lambda\sigma_1 + (1-\lambda)\sigma_2]) \frac{1}{N_q} \varphi(y^x) dy^x \\ &= \sup_{u \in \mathcal{U}} \sum_{y^q} \int_{\mathbb{R}^n} V_{t+1}^*(\lambda\Phi_{y,u}[\sigma_1] + (1-\lambda)\Phi_{y,u}[\sigma_2]) \frac{1}{N_q} \varphi(y^x) dy^x \\ &\leq \sup_{u \in \mathcal{U}} \sum_{y^q} \int_{\mathbb{R}^n} [\lambda V_{t+1}^*(\Phi_{y,u}[\sigma_1]) + (1-\lambda)V_{t+1}^*(\Phi_{y,u}[\sigma_2])] \frac{1}{N_q} \varphi(y^x) dy^x \\ &\leq \sup_{u \in \mathcal{U}} \sum_{y^q} \int_{\mathbb{R}^n} \lambda V_{t+1}^*(\Phi_{y,u}[\sigma_1]) \frac{1}{N_q} \varphi(y^x) dy^x \end{aligned}$$

$$\begin{aligned}
& + \sup_{u \in \mathcal{U}} \sum_{y^q} \int_{\mathbb{R}^n} (1 - \lambda) V_{t+1}^*(\Phi_{y,u}[\sigma_2]) \frac{1}{N_q} \varphi(y^x) dy^x \\
& \leq \lambda V_t^*(\sigma_1) + (1 - \lambda) V_t^*(\sigma_2)
\end{aligned}$$

■

Lemma 2. *For any t , the value function (22) can be written as*

$$V_t^*(\sigma) = \sup_{\alpha_t^i \in \Gamma_t} \langle \alpha_t^i, \sigma \rangle$$

Proof: By induction, at time T

$$V_T^*(\sigma) = \sum_{q \in \mathcal{Q}} \int_{\mathbb{R}^n} \mathbf{1}_K(x, q) \sigma(x, q) dx$$

By defining $\alpha_T(x, q) = \mathbf{1}_K(x, q)$, we obtain the desired result. Note that this definition of α_T is in line with the definition given in Section II-C, because although it does not represent a full policy tree (being at the terminal time, there are no more branches on the tree), it does represent the immediate value of being in state (x, q) , given by $\mathbf{1}_K(x, q)$.

Next, assuming $V_{t+1}^*(\sigma) = \sup_{\Gamma_{t+1}} \langle \alpha_{t+1}^i, \sigma \rangle$, V_t^* can be written as

$$\begin{aligned}
V_t^*(\sigma) &= \sup_{u \in \mathcal{U}} \sum_{y^q} \int_{\mathbb{R}^n} V_{t+1}^*(\Phi_{y,u}[\sigma]) \frac{1}{N_q} \varphi(y^x) dy^x \\
&= \sup_{u \in \mathcal{U}} \sum_{y^q} \int_{\mathbb{R}^n} \sup_{\Gamma_{t+1}} \langle \alpha_{t+1}^i, \Phi_{y,u}[\sigma] \rangle \frac{1}{N_q} \varphi(y^x) dy^x \\
&= \sup_{u \in \mathcal{U}} \sum_{y^q} \int_{\mathbb{R}^n} \sup_{\Gamma_{t+1}} \sum_{q'} \int_{\mathbb{R}^n} \alpha_{t+1}^i(x', q') \Phi_{y,u}[\sigma](x', q') dx' \frac{1}{N_q} \varphi(y^x) dy^x \\
&= \sup_{u \in \mathcal{U}} \sum_{y^q} \int_{\mathbb{R}^n} \sup_{\Gamma_{t+1}} \sum_{q'} \int_{\mathbb{R}^n} \sum_q \int_{\mathbb{R}^n} \alpha_{t+1}^i(x', q') Q_{q', y^q}(u) \varphi(y^x - h(x', u)) \mathbf{1}_K(x, q) \\
&\quad \times \tau(x', q' \mid x, q, u) \sigma(x, q) dx dx' dy^x \\
&= \sup_{u \in \mathcal{U}} \sum_{y^q} \int_{\mathbb{R}^n} \sup_{\Gamma_{t+1}} \sum_q \int_{\mathbb{R}^n} \left[\sum_{q'} \int_{\mathbb{R}^n} \alpha_{t+1}^i(x', q') Q_{q', y^q}(u) \varphi(y^x - h(x', u)) \right. \\
&\quad \left. \times \tau(x', q' \mid x, q, u) dx' \right] \mathbf{1}_K(x, q) \sigma(x, q) dx dy^x
\end{aligned}$$

$$= \sup_{u \in \mathcal{U}} \sum_{y^q} \int_{\mathbb{R}^n} \sup_{\Gamma_{t+1}} \left\langle \sum_{q'} \int_{\mathbb{R}^n} \alpha_{t+1}^i(x', q') Q_{q', y^q}(u) \varphi(y^x - h(x', u)) \right. \\ \left. \times \tau(x', q' \mid x, q, u) dx' \mathbf{1}_K(x, q), \sigma(x, q) \right\rangle dy^x$$

Then for a specific observation y , action u , and α_{t+1}^i function, the function $\alpha_{y,u}^i$ can be defined as

$$\alpha_{y,u}^i(x, q) = \sum_{q'} \int_{\mathbb{R}^n} \alpha_{t+1}^i(x', q') Q_{q', y^q}(u) \varphi(y^x - h(x', u)) \tau(x', q' \mid x, q, u) dx' \mathbf{1}_K(x, q) \quad (25)$$

Because $\alpha_{y,u}^i$ does not depend on σ , we can redefine the supremum over all Γ_{t+1} to be over all $\alpha_{y,u}^i$.

$$V_t^*(\sigma) = \sup_{u \in \mathcal{U}} \sum_{y^q} \int_{\mathbb{R}^n} \sup_{\{\alpha_{y,u}^i\}} \langle \alpha_{y,u}^i, \sigma \rangle dy^x$$

For a specific σ , u , and y , if we define

$$\alpha_{y,u,\sigma}(x, q) = \arg \sup_i \langle \alpha_{y,u}^i, \sigma \rangle \quad (26)$$

then V_t^* can be further simplified as

$$V_t^*(\sigma) = \sup_{u \in \mathcal{U}} \sum_{y^q} \int_{\mathbb{R}^n} \langle \alpha_{y,u,\sigma}, \sigma \rangle dy^x = \sup_{u \in \mathcal{U}} \left\langle \sum_{y^q} \int_{\mathbb{R}^n} \alpha_{y,u,\sigma} dy^x, \sigma \right\rangle$$

Therefore, the set of all $\{\alpha_t^i\}$ can be described by

$$\Gamma_t = \bigcup_{\sigma} \left\{ \sum_{y^q} \int_{\mathbb{R}^n} \alpha_{y,u,\sigma} dy^x \right\}_{\forall u \in \mathcal{U}} \quad (27)$$

and V_t^* may be written as

$$V_t^*(\sigma) = \sup_{\alpha_t^i \in \Gamma_t} \langle \alpha_t^i, \sigma \rangle \quad (28)$$

■

As in [22], for discrete actions and observations, the set Γ_t has finite cardinality, and so $V_t^*(\sigma)$ is a piecewise-linear function in σ . If the state space was small and discrete, as were the observations and actions, we could construct a finite set of σ vectors, and then generate a finite set of α -vectors at each time step k to solve the above problem exactly, much like the algorithm first proposed by [23]. However, with a hybrid state space, there are an infinite number of σ functions defined on an infinite number of states, and so we cannot hope to solve this problem exactly. We can instead sample sufficient statistics σ from the set Σ of all possible

σ functions, just as a collection of sampled belief points are used in [22] and many other PBVI solvers designed for large (but discrete) state spaces. The set of sampled points is denoted $\tilde{\Sigma}$. By sampling from the sufficient statistic space Σ , we can generate a finite number of α -functions. Further, because of the piecewise-linear convex nature of the value functions, we are guaranteed to obtain a lower bound on the true value function. In fact, we can characterize the error between the value functions produced by the point-based method and the true value functions, based on how densely we sample Σ .

The operator H in (23) represents the complete backup operation (17). The operator \tilde{H} will be used to represent a point-based backup based on a set of sampled belief points $\tilde{\Sigma}$. We denote the approximate value function at time t characterized by $\tilde{\Gamma}_t$ as $V_t^{\tilde{\Sigma}}$, in comparison to the true value function V_t^* . Further, let $\delta(\tilde{\Sigma})$ be the maximum L^1 distance between points in $\tilde{\Sigma}$ and points in Σ .

$$\delta(\tilde{\Sigma}) = \sup_{\tilde{\sigma} \in \tilde{\Sigma}} \inf_{\sigma \in \Sigma} \|\tilde{\sigma} - \sigma\|_1 \quad (29)$$

Now consider the maximum error introduced in performing one iteration of point-based backup, given the current value function estimate $V_t^{\tilde{\Sigma}}$.

Lemma 3. *The error introduced in one iteration of point-based value iteration, denoted $\epsilon^{(1)}$, is at most $\delta(\tilde{\Sigma})$:*

$$\left\| \tilde{H}[V_t^{\tilde{\Sigma}}] - H[V_t^{\tilde{\Sigma}}] \right\|_{\infty} = \epsilon^{(1)} \leq \delta(\tilde{\Sigma})$$

Proof: The proof is similar to one in [25] for discrete state POMDPs. First, let $\sigma^{(1)}$ be the point in Σ where the error between the true value function and the point-based backup is greatest. Let $\sigma^{(2)} \in \tilde{\Sigma}$ be the closest point in the L^1 sense to $\sigma^{(1)}$. Let $\alpha^{(2)} \in \tilde{\Gamma}_{t-1}$ be maximal at $\sigma^{(2)}$, and $\alpha^{(1)} \in \Gamma_{t-1}$ (and not in $\tilde{\Gamma}_{t-1}$) is the function that *would* be maximal at $\sigma^{(1)}$ had it been calculated.

Then

$$\begin{aligned} \epsilon^{(1)} &\leq |\langle \alpha^{(1)}, \sigma^{(1)} \rangle - \langle \alpha^{(2)}, \sigma^{(1)} \rangle| \\ &\leq |\langle \alpha^{(1)}, \sigma^{(1)} \rangle - \langle \alpha^{(2)}, \sigma^{(1)} \rangle + \langle \alpha^{(1)}, \sigma^{(2)} \rangle - \langle \alpha^{(1)}, \sigma^{(2)} \rangle| \\ &\leq |\langle \alpha^{(1)}, \sigma^{(1)} \rangle - \langle \alpha^{(2)}, \sigma^{(1)} \rangle + \langle \alpha^{(2)}, \sigma^{(2)} \rangle - \langle \alpha^{(1)}, \sigma^{(2)} \rangle| \\ &\leq |\langle \alpha^{(1)} - \alpha^{(2)}, \sigma^{(1)} - \sigma^{(2)} \rangle| \end{aligned} \quad (30)$$

$$\leq \|\alpha^{(1)} - \alpha^{(2)}\|_\infty \|\sigma^{(1)} - \sigma^{(2)}\|_1 \quad (31)$$

$$\leq \|\alpha^{(1)} - \alpha^{(2)}\|_\infty \delta(\tilde{\Sigma}) \quad (32)$$

Line (30) follows because $\alpha^{(2)}$ is optimal for $\sigma^{(2)}$, implying $\langle \alpha^{(1)}, \sigma^{(2)} \rangle \leq \langle \alpha^{(2)}, \sigma^{(2)} \rangle$. Line (31) follows from Hölder's Inequality. Line (32) can be further simplified by noting that the α -functions are bounded between 0 and 1 for all $x \in \mathcal{X}$ and $q \in \mathcal{Q}$. Because the value function at a specific point σ represents the probability of staying within set K for some length of time, given the normalized density σ , this value must be between 0 and 1. The value function is further defined as $\sup \langle \alpha, \sigma \rangle$, meaning that the inner product of α and σ must be between 0 and 1, and therefore α must be between 0 and 1 for all x, q (since by (25) it clearly must be nonnegative).

Therefore, we can say $\|\alpha^{(1)} - \alpha^{(2)}\|_\infty \leq 1$, and we get that $\epsilon^{(1)} \leq \delta(\tilde{\Sigma})$. ■

We now use Lemma 3 to derive a bound between the true value function and the point-based approximation at any time t .

Theorem 1. *For a set of sufficient statistics Σ , sampled set $\tilde{\Sigma}$, and horizon t , the error from using point-based value iteration versus full value iteration, given by $\epsilon(t) = \|V_t^{\tilde{\Sigma}} - V_t^*\|_\infty$ is bounded above by*

$$\|V_t^{\tilde{\Sigma}} - V_t^*\|_\infty = \epsilon(t) \leq t\delta(\tilde{\Sigma})$$

Proof:

$$\begin{aligned} \epsilon(t) &= \|V_{T-t}^{\tilde{\Sigma}} - V_{T-t}^*\|_\infty \\ &= \|\tilde{H}[V_{T-t-1}^{\tilde{\Sigma}}] - H[V_{T-t-1}^*]\|_\infty \\ &= \|\tilde{H}[V_{T-t-1}^{\tilde{\Sigma}}] - H[V_{T-t-1}^*] + H[V_{T-t-1}^{\tilde{\Sigma}}] - H[V_{T-t-1}^{\tilde{\Sigma}}]\|_\infty \\ &\leq \|\tilde{H}[V_{T-t-1}^{\tilde{\Sigma}}] - H[V_{T-t-1}^{\tilde{\Sigma}}]\|_\infty + \|H[V_{T-t-1}^{\tilde{\Sigma}}] - H[V_{T-t-1}^*]\|_\infty \\ &\leq \epsilon^{(1)} + \|V_{T-t-1}^{\tilde{\Sigma}} - V_{T-t-1}^*\|_\infty \end{aligned} \quad (33)$$

$$\leq \epsilon^{(1)} + \epsilon(t-1)$$

$$\epsilon(t) \leq t\delta(\tilde{\Sigma}) \quad (34)$$

Line (33) follows from the definition of $\epsilon^{(1)}$, and line (34) follows from Lemma 3. ■

Thus the error between the point-based approximation and the actual value function is directly proportional to how densely $\tilde{\Sigma}$ is sampled, and converges to zero as $\tilde{\Sigma}$ approaches Σ .

B. Implementation

For a state space \mathcal{S} that is discrete, “closedness” of the belief state $b(s)$ and of the α -vectors $\alpha_t(s)$ is maintained after updates by the operator $M_{y,u}$ and by (13)-(14), respectively. That is, although the belief function can take on an infinite number of values for each state s (the interval $[0, 1]$), because there are a finite number of states in \mathcal{S} , the function $b(s)$ can be represented by a vector $[b(s_0) b(s_1) \dots b(s_n)]$ with each entry $b(s_i) \in [0, 1]$ corresponding to the probability of being in state s_i according to the specific density b . Similarly for the set of α -vectors, Γ_t , which remain the same size after updates according to (13) and (14).

For \mathcal{S} continuous, this “closedness” property of the structure of both the beliefs and α -functions under updating is no longer guaranteed, and can make the computation intractable. As a remedy, [22] represents both the beliefs and α -functions as sums of weighted Gaussians (which can represent a function to any desired accuracy with enough components), and shows that for an additive cost POMDP, the belief function remains a Gaussian sum under the belief update operator $M_{y,u}$, as do the α -functions when generated recursively from the previous set of α -functions. The Gaussian sum representation also guarantees the inner product operation $\langle \alpha, b \rangle$ to be computable.

We now show that we can approximate the sufficient statistic σ by a vector whose entries are finite sums of Gaussians (each entry of the vector corresponds to a different discrete mode q), and that this representation is closed under the update operator Φ . We also show that the α -functions as defined by (27) for the multiplicative reachability cost function can also be approximated by vectors of finite sums of Gaussians, and are closed under the operations defined in (25) and (26). All of the following derivations assume a discrete observation space of finite cardinality $N_{y^q} \times N_{y^x}$. We make the following additional assumptions:

Assumption 1: We can represent the indicator function as a finite sum of Gaussians (35), with $w_i(q) \in \mathbb{R}$ a mode-dependent coefficient, such that for $q \in K_q$, $w_i(q) = 1$, and for $q \notin K_q$, $w_i(q) = 0$ for all i , where $K = K_x \times K_q$. Gaussian distribution i has mean μ_i and covariance

Σ_i .

$$\mathbf{1}_K(x, q) \approx \sum_{i=1}^I w_i(q) \phi(x; \mu_i, \Sigma_i) \quad (35)$$

Assumption 2: We can approximate the stochastic kernel $\tau(s' | s, u) = T_x(x' | x, q', u) T_q(q' | x, q, u)$ by a Gaussian sum. We first express the distribution of the discrete variable q' in terms of Gaussian distributions evaluated at the continuous variable x :

$$T_q(q' | x, q, u) \approx \sum_{j=1}^J w_j(q', q, u) \phi(x; \mu_j(q', q, u), \Sigma_j(q', q, u)) \quad (36)$$

For finite J (36) will never exactly sum to 1 (see [24]) and so will always be an approximation. We assume that the continuous dynamics are linear in x with Gaussian noise, so that

$$T_x(x' | x, q', u) = \phi(x'; \mu_{q'}^u(x), \mathcal{W}_{q'}^u) \quad (37)$$

where $\mu_{q'}^u(x)$ is of the form $Ax + f(q', u)$ with $A \in \mathbb{R}^{n \times n}$ invertible and f a possibly non-linear function of q' and u . This allows us to rewrite T_x in terms of x rather than x' , so that $T_x(x' | x, q', u) = \delta \phi(x; \hat{\mu}_{q'}^u(x'), \hat{\mathcal{W}}_{q'}^u)$ as well. In fact,

$$\phi(x'; Ax + f(q', u), \mathcal{W}_{q'}^u) = |A^{-1}| \phi(x; A^{-1}(x' - f(q', u)), A^{-1} \mathcal{W}_{q'}^u (A^{-1})^T) \quad (38)$$

with $\hat{\mu}_{q'}^u(x') = A^{-1}(x' - f(q', u))$ and $\hat{\mathcal{W}}_{q'}^u = A^{-1} \mathcal{W}_{q'}^u (A^{-1})^T$.

Assumption 3: The discrete observation model for the continuous variable, $\varphi(y^x | x, u)$ can be approximated by

$$\varphi(y^x | x, u) \approx \sum_{h=1}^H w_h(y, u) \phi(x; \mu_h(y, u), \Sigma_h(y, u)) \quad (39)$$

To make notation (slightly) cleaner, we now shift any parameter's dependence on either y or u to its superscript, and any dependence on q or q' to its subscript, so for instance $w_h(y, u)$ becomes $w_h^{y,u}$ and $\mu_j(q', q, u)$ becomes $\mu_{j,q'}^u$.

C. Approximating the Sufficient Statistic

Lemma 4. *The sufficient statistic $\sigma_t(x, q)$ can be approximated by a linear combination of Gaussians for all $t = 0, 1, \dots$, where the parameters of each Gaussian component are dependent on the discrete variable q .*

$$\sigma_t(x, q) \approx \sum_{l=1}^L w_{l,q} \phi(x; \mu_{l,q}, \Sigma_{l,q}) \quad (40)$$

Proof: The proof follows by induction. For $t = 0$, $\sigma_0(x, q) = \rho(x, q)$. Because any distribution can be approximated to arbitrary accuracy by a weighted sum of Gaussians, we set $\rho(x, q) = \sum_{l=1}^L w_{l,q} \phi(x; \mu_{l,q}, \Sigma_{l,q})$ and so $\sigma_0(x, q)$ is of the form (40).

For $t = n - 1$, assume that $\sigma_{n-1}(x, q) = \sum_{l=1}^L w_{l,q} \phi(x; \mu_{l,q}, \Sigma_{l,q})$. Then under the operator Φ given by (21), it follows that

$$\begin{aligned}
\sigma_n(x', q') &= N_{y^q} N_{y^x} Q_{q', y^q}(u) \psi(y^x \mid x', u) \sum_{q=1}^{N_q} \int_{\mathbb{R}^n} \mathbf{1}_K(x, q) T_x(x' \mid x, q', u) T_q(q' \mid q, x, u) \sigma_{n-1}(x, q) dx \\
&\approx N_{y^q} N_{y^x} Q_{q', y^q}(u) \left[\sum_{h=1}^H w_h^{y,u} \phi(x'; \mu_h^{y,u}, \Sigma_h^{y,u}) \right] \sum_{q=1}^{N_q} \int_{\mathbb{R}^n} \left[\sum_{i=1}^I w_{i,q} \phi(x; \mu_i, \Sigma_i) \right] \\
&\quad \times |A^{-1}| \phi(x; A^{-1}(x' - f(q', u)), A^{-1} \mathcal{W}_{q'}^u (A^{-1})^T) \left[\sum_{j=1}^J w_{j,q,q'}^u \phi(x; \mu_{j,q,q'}^u, \Sigma_{j,q,q'}^u) \right] \\
&\quad \times \left[\sum_{l=1}^L w_{l,q} \phi(x; \mu_{l,q}, \Sigma_{l,q}) \right] dx \\
&\approx \sum_{h=1}^H \sum_{i=1}^I \sum_{j=1}^J \sum_{l=1}^L \sum_{q=1}^{N_q} N_{y^q} N_{y^x} Q_{q', y^q}(u) |A^{-1}| w_h^{y,u} w_{i,q} w_{j,q,q'}^u w_{l,q} \phi(x'; \mu_h^{y,u}, \Sigma_h^{y,u}) \\
&\quad \times \int_{\mathbb{R}^n} \phi(x; \mu_i, \Sigma_i) \phi(x; \hat{\mu}_{q'}^u(x'), \hat{\mathcal{W}}_{q'}^u) \phi(x; \mu_{j,q,q'}^u, \Sigma_{j,q,q'}^u) \phi(x; \mu_{l,q}, \Sigma_{l,q}) dx
\end{aligned}$$

Next, the below identity regarding multiplied Gaussians is used to combine the above Gaussians inside the integral.

$$\begin{aligned}
\phi(x; \mu_1, \Sigma_1) \phi(x; \mu_2, \Sigma_2) &= \phi(\mu_1; \mu_2, \Sigma_1 + \Sigma_2) \phi(x; \tilde{\mu}, \tilde{\Sigma}) \\
\tilde{\mu} &= \tilde{\Sigma}(\Sigma_1^{-1} \mu_1 + \Sigma_2^{-1} \mu_2) \\
\tilde{\Sigma} &= (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}
\end{aligned} \tag{41}$$

Then

$$\begin{aligned}
\sigma_n(x', q') &\approx \sum_{h,i,j,l,q} N_{y^q} N_{y^x} Q_{q', y^q}(u) |A^{-1}| w_h^{y,u} w_{i,q} w_{j,q,q'}^u w_{l,q} \phi(x'; \mu_h^{y,u}, \Sigma_h^{y,u}) \phi(\hat{\mu}_{q'}^u(x'); \mu_i, \hat{\mathcal{W}}_{q'}^u + \Sigma_i) \\
&\quad \times \phi(\mu_{j,q,q'}^u; \mu_{l,q}, \Sigma_{j,q,q'}^u + \Sigma_{l,q}) \int_{\mathbb{R}^n} \phi(x; \tilde{\mu}_1(x'), \tilde{\Sigma}_1) \phi(x; \tilde{\mu}_2, \tilde{\Sigma}_2) dx
\end{aligned}$$

with

$$\begin{aligned}\tilde{\mu}_1(x') &= \tilde{\Sigma}_1 \left(\Sigma_i^{-1} \mu_i + \left(\hat{\mathcal{W}}_{q'}^u \right)^{-1} \hat{\mu}_{q'}^u(x') \right), \quad \tilde{\Sigma}_1 = \left(\Sigma_i^{-1} + \left(\hat{\mathcal{W}}_{q'}^u \right)^{-1} \right)^{-1} \\ \tilde{\mu}_2 &= \tilde{\Sigma}_2 \left(\left(\Sigma_{j,q,q'}^u \right)^{-1} \mu_{j,q,q'}^u + \Sigma_{l,q}^{-1} \mu_{l,q} \right), \quad \tilde{\Sigma}_2 = \left(\left(\Sigma_{j,q,q'}^u \right)^{-1} + \Sigma_{l,q}^{-1} \right)^{-1}\end{aligned}$$

using (41). Multiplying the final two Gaussians inside the integral leaves only one Gaussian that is a function of x , which integrates to 1, leaving

$$\begin{aligned}\sigma_n(x, q) &\approx \sum_{h,i,j,l,q} N_{y^q} N_{y^x} Q_{q',y^q}(u) |A^{-1}| w_h^{y,u} w_{i,q} w_{j,q,q'}^u w_{l,q} \phi(\mu_{j,q,q'}^u; \mu_{l,q}, \Sigma_{j,q,q'}^u + \Sigma_{l,q}) \phi(x'; \mu_h^{y,u}, \Sigma_h^{y,u}) \\ &\quad \times \phi(\hat{\mu}_{q'}^u(x'); \mu_i, \hat{\mathcal{W}}_{q'}^u + \Sigma_i) \phi(\tilde{\mu}_1(x'); \tilde{\mu}_2, \tilde{\Sigma}_1 + \tilde{\Sigma}_2)\end{aligned}$$

Now all that is left to complete the proof is to manipulate the last two Gaussians, $\phi(\hat{\mu}_{q'}^u(x'); \mu_i, \hat{\mathcal{W}}_{q'}^u + \Sigma_i)$ and $\phi(\tilde{\mu}_1(x'); \tilde{\mu}_2, \tilde{\Sigma}_1 + \tilde{\Sigma}_2)$ so that they are functions of x' , i.e. $\phi(x'; \hat{\mu}, \hat{\Sigma})$, and then apply (41) twice. This can be done in both cases using straightforward but tedious linear algebra.

First noting that

$$\begin{aligned}\phi(\hat{\mu}_{q'}^u(x'); \mu_i, \hat{\mathcal{W}}_{q'}^u + \Sigma_i) &= |A| \phi(x'; A\mu_i + f(q', u), \mathcal{W}_{q'}^u + A\Sigma_i A^T) \\ \phi(\tilde{\mu}_1(x'); \tilde{\mu}_2, \tilde{\Sigma}_1 + \tilde{\Sigma}_2) &= |A \hat{\mathcal{W}}_{q'}^u \tilde{\Sigma}_1^{-1}| \phi(x'; \bar{\mu}_{q,q'}^u, \bar{\Sigma}_{q,q'}^u) \\ \bar{\mu}_{q,q'}^u &= A \left[\hat{\mathcal{W}}_{q'}^u \tilde{\Sigma}_1^{-1} (\tilde{\mu}_2 - \tilde{\Sigma}_1 \Sigma_i^{-1} \mu_i) + f(q', u) \right] \\ \bar{\Sigma}_{q,q'}^u &= A \left[\hat{\mathcal{W}}_{q'}^u \tilde{\Sigma}_1^{-1} (\tilde{\Sigma}_1 + \tilde{\Sigma}_2) \hat{\mathcal{W}}_{q'}^u \tilde{\Sigma}_1^{-1} \right] A^T\end{aligned}$$

we can ultimately write

$$\begin{aligned}\sigma_n(x', q') &\approx \sum_{h,i,j,l,q} w_{h,i,j,l,q,q'} \phi(x'; \mu_{h,i,j,l,q,q'}, \Sigma_{h,i,j,l,q,q'}) \\ &\quad \text{HIJLN}_q \\ &\approx \sum_{k=1} w_{k,q'} \phi(x'; \mu_{k,q'}, \Sigma_{k,q'})\end{aligned}\tag{42}$$

where

$$\begin{aligned}w_{h,i,j,l,q,q'} &= |A \hat{\mathcal{W}}_{q'}^u \tilde{\Sigma}_1^{-1}| N_{y^q} N_{y^x} Q_{q',y^q}(u) w_h^{y,u} w_{i,q} w_{j,q,q'}^u w_{l,q} \phi(\mu_{j,q,q'}^u; \mu_{l,q}, \Sigma_{j,q,q'}^u + \Sigma_{l,q}) \\ &\quad \times \phi(\mu_h^{y,u}; A\mu_i + f(q', u), \Sigma_h^{y,u} + \mathcal{W}_{q'}^u + A\Sigma_i A^T) \phi(\bar{\mu}_{q,q'}^u; c, C + \bar{\Sigma}_{q,q'}^u)\end{aligned}\tag{43}$$

$$\mu_{h,i,j,l,q,q'} = \left(C^{-1} + (\bar{\Sigma}_{q,q'}^u)^{-1} \right)^{-1} \left(C^{-1} c + (\bar{\Sigma}_{q,q'}^u)^{-1} \bar{\mu}_{q,q'}^u \right)\tag{44}$$

$$\Sigma_{h,i,j,l,q,q'} = \left(C^{-1} + (\bar{\Sigma}_{q,q'}^u)^{-1} \right)^{-1}\tag{45}$$

$$C = \left((\Sigma_h^{y,u})^{-1} + (\mathcal{W}_{q'}^u + A\Sigma_i A^T)^{-1} \right)^{-1} \quad (46)$$

$$c = C \left((\Sigma_h^{y,u})^{-1} \mu_h^{y,u} + (\mathcal{W}_{q'}^u + A\Sigma_i A^T)^{-1} (A\mu_i + f(q', u)) \right) \quad (47)$$

■

The sufficient statistic σ is therefore closed under the update operator Φ . The expression in (42) - (47) simplifies somewhat depending on the problem, as seen in Section IV. More problematic is the explosion in the number of Gaussians: for L Gaussians representing σ_{n-1} , $HIJLN_{y^q}$ Gaussians are required to represent σ_n . However, there are techniques to combine similar components (the individual weighted Gaussians) of the mixture in order to bound the total components, which will be discussed in Section IV.

D. Approximating the α -Functions

We use the same approach as in Lemma 4 to approximate the α -functions by Gaussian mixtures, through induction and application of the operation defined in (25) that generates $\alpha_{y,u}^i$ from α_{n+1}^i . Showing that (25) preserves the Gaussian mixture structure of the α -functions is sufficient to show that the full *backup* operation is closed under Gaussian sums when the observations y^x are discrete, since the only additional operation is to sum over all y^x , as in (27).

Lemma 5. *The α -functions $\alpha_t^i(x, q)$ can be approximated by a linear combination of Gaussians for all $t = 0, 1, \dots$ where the parameters for each Gaussian are dependent on the discrete variable q .*

$$\alpha_t^i(x, q) \approx \sum_{d=1}^D w_{d,q} \phi(x; \mu_{d,q}, \Sigma_{d,q}) \quad (48)$$

Proof: We omit most details of the proof, since they are almost identical to those in the proof of Lemma 4.

For $t = T$, from Lemma 2 and the definition of α_T as the indicator function $\mathbf{1}_K(s)$, setting

$$\alpha_T(x, q) \approx \sum_{i=1}^I w_{i,q} \phi(x; \mu_i, \Sigma_i)$$

using the Gaussian sum approximation to the indicator function (35) gives α_T in the desired form.

Assuming $\alpha_{n+1}^j(x', q') = \sum_{d=1}^D w_{d,q'} \phi(x'; \mu_{d,q'}, \Sigma_{d,q'})$, using (25) it follows that

$$\begin{aligned}
\alpha_{y,u}^j(x, q) &\approx \sum_{q'=1}^{N_q} Q_{q',y^q}(u) \int_{\mathbb{R}^n} \left[\sum_{d=1}^D w_{d,q'} \phi(x'; \mu_{d,q'}, \Sigma_{d,q'}) \right] \left[\sum_{h=1}^H w_h^{y,u} \phi(x'; \mu_h^{y,u}, \Sigma_h^{y,u}) \right] \\
&\quad \times \phi(x'; \mu_{q'}^u(x), \mathcal{W}_{q'}^u) dx' \left[\sum_{j=1}^J w_{j,q',q}^u \phi(x; \mu_{j,q',q}^u, \Sigma_{j,q',q}^u) \right] \left[\sum_{i=1}^I w_{i,q} \phi(x; \mu_i, \Sigma_i) \right] \\
&\approx \sum_{q',d,h,j,i} w_{q',d,h,j,i,q} \phi(x; \mu_{q',d,h,j,i,q}, \Sigma_{q',d,h,j,i,q})
\end{aligned}$$

where

$$\begin{aligned}
w_{q',d,h,j,i,q} &= |A^{-1}| Q_{q',y^q}(u) w_{d,q'} w_h^{y,u} w_{j,q',q}^u w_{i,q} \phi(\mu_i; \mu_{j,q',q}^u, \Sigma_i + \Sigma_{j,q',q}^u) \phi(\mu_{d,q'}; \mu_h^{y,u}, \Sigma_{d,q'} + \Sigma_h^{y,u}) \\
&\quad \times \phi(\tilde{\mu}_1; A^{-1}(\tilde{\mu}_2 - f(q', u)), \tilde{\Sigma}_1 + A^{-1}(\tilde{\Sigma}_2 + \mathcal{W}_{q'}^u) (A^{-1})^T)
\end{aligned} \tag{49}$$

$$\mu_{q',d,h,j,i,q} = \left(A^T (\tilde{\Sigma}_2 + \mathcal{W}_{q'}^u)^{-1} A + \tilde{\Sigma}_1^{-1} \right)^{-1} \left(A^T (\tilde{\Sigma}_2 + \mathcal{W}_{q'}^u)^{-1} (\tilde{\mu}_2 - f(q', u)) + \tilde{\Sigma}_1^{-1} \tilde{\mu}_1 \right) \tag{50}$$

$$\Sigma_{q',d,h,j,i,q} = \left(A^T (\tilde{\Sigma}_2 + \mathcal{W}_{q'}^u)^{-1} A + \tilde{\Sigma}_1^{-1} \right)^{-1} \tag{51}$$

$$\begin{aligned}
\tilde{\mu}_1 &= \tilde{\Sigma}_1 \left((\Sigma_{j,q',q}^u)^{-1} \mu_{j,q',q}^u + \Sigma_i^{-1} \mu_i \right), \quad \tilde{\Sigma}_1 = \left((\Sigma_{j,q',q}^u)^{-1} + \Sigma_i^{-1} \right)^{-1} \\
\tilde{\mu}_2 &= \tilde{\Sigma}_2 \left(\Sigma_{d,q'}^{-1} \mu_{d,q'} + (\Sigma_h^{y,u})^{-1} \mu_h^{y,u} \right), \quad \tilde{\Sigma}_2 = \left(\Sigma_{d,q'}^{-1} + (\Sigma_h^{y,u})^{-1} \right)^{-1}
\end{aligned}$$

Because each $\alpha_{y,u}^j(x, q)$ is approximated by a sum of Gaussians for all j , any $\alpha_n^j \in \tilde{\Gamma}_n$ is also a sum of Gaussians (since the only additional operation to generate the α_n^j from $\alpha_{y,u}^j$ is to sum over all y^x and y^q , as in (27)). ■

IV. EXAMPLE

The temperature regulation problem is a benchmark example for hybrid systems, and a stochastic version with perfect state information is presented in [28]. We consider the case of one heater, which can either be turned on to heat one room, or turned off. The temperature of the room at time t is given by the continuous variable $x(t)$, and the discrete state $q(t) = 1$ indicates the heater is on at time t , and $q(t) = 0$ denotes the heater is off. The stochastic difference equation governing the temperature is given by

$$x(t+1) = (1-b)x(t) + cq(t+1) + bx_a + v(t)$$

with constants $b = 0.0167$, $c = 0.8$, and $x_a = 6$, and $v(t)$ i.i.d. Gaussian random variables with mean zero and variance v^2 . The control input is given by $u(t) \in \mathcal{U}$ with $\mathcal{U} = \{0, 1\}$, but the chosen control is not always implemented with probability 1. Instead, $q(t)$ is updated probabilistically, dependent on $u(t-1)$ and $q(t-1)$, with transition function $T_q(q(t+1) | q(t), u(t))$. So while function $\bar{\mu}_t(\sigma_t)$ deterministically returns a single control input, control input $u_t = \bar{\mu}_t(\sigma_t)$ may not always be implemented.

To model this as a partially observable problem, assume the actual temperature is unknown, and only a noisy measurement is available to the controller. The controller does, however, know if the heater is on or off at time t (i.e. $q(t)$ is perfectly observed). The observation $y(t) = y^x(t)$ is given by $y^x(t) = x(t) + w(t)$, with $w(t)$ i.i.d. Gaussian random variables with mean zero and variance w^2 (so that $\varphi(\hat{w}) = \phi(\hat{w}; 0, w^2)$). Because the discrete mode q is perfectly observed, we do not record $y^q(t)$, and it is not included in any equations.

It is desirable to keep the temperature of the room between 17.5 and 22 degrees celsius at all times, hence the safe region $K = [17.5, 22]$ does not depend on the discrete state $q(k)$ (so $\mathbf{1}_K(s) = \mathbf{1}_K(x)$). To find the maximum probability that the room stays within the desired temperature range given that the controller only has access to the mode $q(t)$ and observations $y^x(t)$, we first find expressions for both $\sigma_t(x, q)$ and $\alpha_t(x, q)$ as Gaussian sums.

We first discretize the observations y^x . Using a grid with spacing Δ , y^x is redefined over $\mathcal{Y} = \{17.5 - \text{tol}_1, 17.5 - \text{tol}_1 + \Delta, \dots, 22 + \text{tol}_2 - \Delta, 22 + \text{tol}_2\}$ where tol_1 and tol_2 are defined so that the probability of observing y^x outside of $[17.5 - \text{tol}_1, 22 + \text{tol}_2]$ is approximately zero. The probability that $y^x = \bar{y} \in \mathcal{Y}$ can be written as

$$\varphi(\bar{y} - x) = \mathbb{P} \left[y^x \in \left[\bar{y} - \frac{\Delta}{2}, \bar{y} + \frac{\Delta}{2} \right] \middle| x \right] = \int_{\bar{y} - \frac{\Delta}{2}}^{\bar{y} + \frac{\Delta}{2}} \phi(y; x, w^2) dy$$

which in turn can be approximated by a summation:

$$\int_{\bar{y} - \frac{\Delta}{2}}^{\bar{y} + \frac{\Delta}{2}} \phi(y; x, w^2) dy = \int_{\bar{y} - \frac{\Delta}{2}}^{\bar{y} + \frac{\Delta}{2}} \phi(x; y, w^2) dy \approx \sum_{h=\bar{y} - \frac{\Delta}{2}}^{\bar{y} + \frac{\Delta}{2} - \delta_H} \delta_H \phi(x; h, w^2)$$

For δ_H the grid spacing in the interval $[\bar{y} - \frac{\Delta}{2}, \bar{y} + \frac{\Delta}{2}]$, we can now write the discretized observation function as a sum of Gaussians:

$$\varphi(y^x - x) \approx \sum_{h=1}^H w_h \phi(x; \mu_h(y^x), w^2) \quad (52)$$

with $w_h = \delta_H$ for all h , and $\mu_h(y^x) = y^x - \frac{\Delta}{2} + (h-1)\delta_H$.

The one dimensional indicator function also needs to be approximated by a sum of Gaussians. Unfortunately, because the indicator function is discontinuous, approximation by a finite sum of Gaussians induces a pseudo-Gibbs phenomenon, with oscillations occurring near the discontinuities (endpoints of K). Using more components leads to a smoother approximation in the interior of K , but the oscillations at the endpoints remain. Unfortunately, no clear ways to avoid this phenomenon currently exist. It should be noted that the inability to exactly represent the indicator function using a Gaussian mixture leads to α -functions that are also only approximations to the true α -functions, and thus the guaranteed lower bound on the value function breaks down. One practical workaround is to choose Gaussian components that slightly underapproximate the indicator function (except at the endpoints), to help preserve the underapproximation to the true value function. For low-dimensional problems we have not experienced any problems approximating the indicator function and obtaining a reasonable lower bound for the value function, but at higher dimensions it is possible that the number of Gaussian components required for reasonable approximations becomes prohibitive.

A recursive expression for $\sigma_k(x, q)$ can be found using the derivation given in section III-C. For an initial distribution $\rho(x)$ on x_0 that is Gaussian with mean μ_0 and variance s^2 , and for $q_0 = 0$, then

$$\sigma_0(x, q) = \mathbf{1}_{\{0\}}(q)\rho(x) = \begin{bmatrix} \sigma_0(x, 0) \\ \sigma_0(x, 1) \end{bmatrix} = \begin{bmatrix} \phi(x; \mu_0, s^2) \\ 0 \end{bmatrix} \quad (53)$$

Given we already have an approximation to $\sigma_t(x, q)$ as in (40), σ_{t+1} corresponding to observation y and control input u can be written as

$$\sigma_{t+1}(x, q) = \sum_{q_t=1}^{N_q} \sum_{i=1}^I \sum_{h=1}^H \sum_{l=1}^L w_{i,h,l,q_t}(q, u, y) \phi(x; \mu_{i,h,l,q_t}(q, u, y), \Sigma_{i,h,l}) \quad (54)$$

with

$$\begin{aligned} w_{i,h,l,q_t}(q, u, y) &= N_{y^x} T_q(q \mid q_k, u) w_i w_{l,q_t} w_h \phi(\mu_i; \mu_{l,q_t}, \Sigma_l + \Sigma_i) \\ &\quad \times \phi(\mu_h(y); cq + bx_a + (1-b)\hat{\mu}, w^2 + \hat{\Sigma}(1-b)^2 + v^2) \\ \mu_{i,h,l,q_t}(q, y) &= \frac{\mu_h(y) \left(\hat{\Sigma}(1-b)^2 + v^2 \right) + (cq + bx_a + (1-b)\hat{\mu}) w^2}{w^2 + v^2 + \hat{\Sigma}(1-b)^2} \end{aligned}$$

$$\Sigma_{i,h,l} = \frac{w^2 \left(\hat{\Sigma}(1-b)^2 + v^2 \right)}{w^2 + v^2 + \hat{\Sigma}(1-b)^2}$$

$$\hat{\mu} = \frac{\mu_i \Sigma_l + \mu_{l,q_t} \Sigma_i}{\Sigma_l + \Sigma_i}, \quad \hat{\Sigma} = \frac{\Sigma_l \Sigma_i}{\Sigma_l + \Sigma_i}$$

Likewise, given $\alpha_{t+1}^j(x, q)$ in the form (48), $\alpha_{y,u}^j(x, q)$ corresponding to observation y and control u can be written

$$\alpha_{y,u}^j(x, q) = \sum_{q_{t+1}=1}^{N_q} \sum_{i=1}^I \sum_{h=1}^H \sum_{l=1}^L w_{i,h,l,q_{t+1}}(q, u, y) \phi(x; \mu_{i,h,l,q_{t+1}}(y), \Sigma_{i,h,l,q_{t+1}})$$

with

$$w_{i,h,l,q_{t+1}}(q, u, y) = \frac{1}{1-b} T_q(q_{t+1} | q, u) w_i w_{l,q_{t+1}} w_h \phi(\mu_h(y); \mu_l, \Sigma_l + w^2)$$

$$\times \phi \left(\mu_i; \frac{\hat{\mu} - cq_{t+1} - bx_a}{1-b}, \Sigma_i + \frac{\hat{\Sigma} + v^2}{(1-b)^2} \right)$$

$$\mu_{i,h,l,q_{t+1}}(y) = \frac{\mu_i(\hat{\Sigma} + v^2) + (1-b)\Sigma_i(\hat{\mu} - cq_{t+1} - bx_a)}{\hat{\Sigma} + v^2 + (1-b)^2\Sigma_i}$$

$$\Sigma_{i,h,l,q_{t+1}} = \frac{\Sigma_i(v^2 + \hat{\Sigma})}{\hat{\Sigma} + v^2 + (1-b)^2\Sigma_i}$$

$$\hat{\mu} = \frac{\mu_h(y)\Sigma_l + \mu_l w^2}{w^2 + \Sigma_l}, \quad \hat{\Sigma} = \frac{w^2 \Sigma_l}{w^2 + \Sigma_l}$$

We implemented an algorithm in the style of POMDP solver Perseus [22] to generate an approximation to the value functions, by generating a fixed set of belief points, which we backed up at each iteration. Unlike Perseus, we backed up every belief point, as necessary for a finite horizon calculation. To generate the set $\tilde{\Sigma}$ of belief states, we first generated a set of initial distributions $\rho(x) = \phi(x; \mu_0, s^2)$ by fixing the variance to be $s^2 = 0.1$, and uniformly selecting the mean μ_0 at random within the values of 17.5 and 22. We then randomly sampled observations y^x uniformly on $[16, 23.5]$, and chose an action $u \in \{0, 1\}$ at random as well. We continued this process for each σ for T time steps, and updated each σ accordingly using (54). If a σ function came too close to being everywhere zero, we reset that σ to a new σ_0 and began the process again.

We also used a mixture reduction process described in [29] to combine similar Gaussian components into a single new Gaussian based on the L^2 distance between the functions. Once a new α -function or σ was generated from the previous α -function or σ , we used the algorithm

in [29] to reduce the total number of Gaussians to 20. This helped reduce computation time, without overly sacrificing accuracy. The number of components to keep can be easily changed, however, depending on the importance of trade-offs in speed versus accuracy.

Using a set $\tilde{\Sigma}$ of 40 σ s, and running the backup operation T times, we obtained an estimate of the probability of the temperature remaining within set K for various μ_0 values, and $q(0) = 0$. We also used the α -functions calculated in the T th iteration as a stationary policy to estimate the average reachability probability for various μ_0 . To do so, we ran 200 simulations of the temperature of the room over T time steps for each μ_0 , using the stationary policy generated by the α -functions to choose control actions. The results of both the approximation to the probabilities via the value function estimate, as well as the probability estimated through simulation of the policy, are presented in Fig. 1 for $T = 5$ (1a) and $T = 20$ (1b).

The value function estimate of the probability closely resembles the estimates from simulation, although near the edges of K the discrepancies are larger. This is likely due to the inaccuracies in the Gaussian sum approximation to the indicator function, which are much more noticeable at the boundaries. The α -functions also consistently produce lower probabilities than the simulated optimal policy, partly due to the inaccuracy of the indicator function approximation, but also because they are designed to produce a lower bound on the true value function. Fig. 2 shows the optimal choice of u_0 according to the α -functions for varying μ_0 (i.e. for varying σ_0 , since $\sigma_0 \sim \mathcal{N}(\mu_0, s^2)$). When the mean μ_0 is less than or equal to 19.3, the heater should be turned on ($u_0 = 1$), and for larger values of μ_0 the heater should remain off.

Computation time to produce the α -functions is intensive. For $T = 20$ time steps, generating the α -functions took approximately eight hours to calculate on an Intel 3.40 GHz CORE i7-2600 CPU with 8 GB of RAM. However, once the α -functions have been calculated, using them to generate optimal control actions takes less than a second, including the time required to update the belief. Thus, to estimate the probability of remaining in K for a single sample trajectory, both when generating the σ functions as Gaussian sums, and finding the optimal α -function and associated control input, only takes a few seconds.

We found that the main bottleneck in computation was the discretized observations. The PBVI algorithm requires evaluating the set of all observations at several different times in the backup process, and hence is not well suited to a large number of discrete observations. Another issue in extending this method to higher dimensional systems is in approximating the indicator function

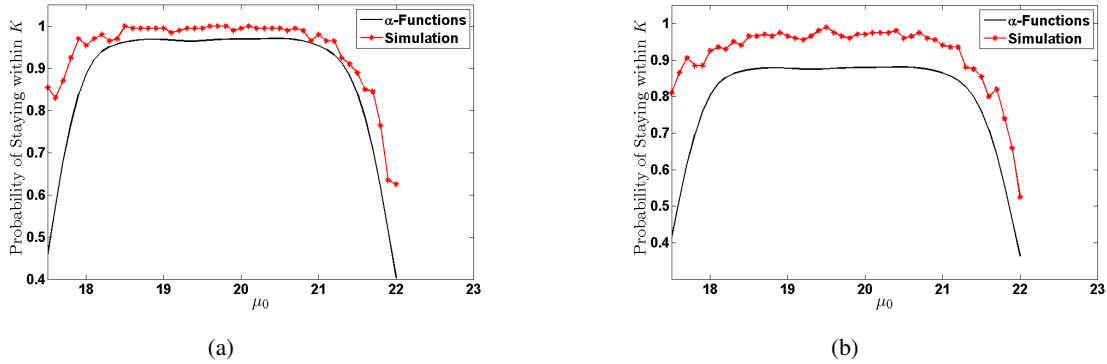


Fig. 1: Given $\sigma_0(x_0, q_0) = \mathbf{1}_{\{0\}}(q_0)\phi(x_0; \mu_0, 0.1)$, estimated probability of x_k staying in $[17.5, 22]$ for (a) $T = 5$ time steps and (b) $T = 20$ time steps according to α -functions (in black) and according to simulation that uses the policy from the α -functions (in red). The α -functions consistently underestimate the simulated reachability probability, assuring a minimum probability of safety, although the estimates from both have the same behavior, and are not too different except towards the boundaries of K .

as a sum of Gaussians, which is required both in the α -function representation and in the belief update. The growth in the number of Gaussians needed to adequately approximate a higher dimensional indicator function slows down the overall computation time. Therefore in order to apply this PBVI technique to higher dimensional systems, we will need to explore better representations of the indicator function (possibly using a particle filter to represent the beliefs, as in [22] or [21]) as well as efficient ways to allow for continuous observations, possibly by using the method described in [22], which groups observations according to which α -functions are optimal for those observations and creating a discrete representative for each group.

V. CONCLUSION

We have provided the first numerical results to the reachability problem with partially observable discrete time stochastic hybrid dynamics. By showing that the value function is still piecewise-linear and convex in the case of discrete actions and observations, and that the representation of the α -functions and belief states by linear combinations of Gaussians is preserved under the backup operator and belief update, we were able to extend PBVI techniques for

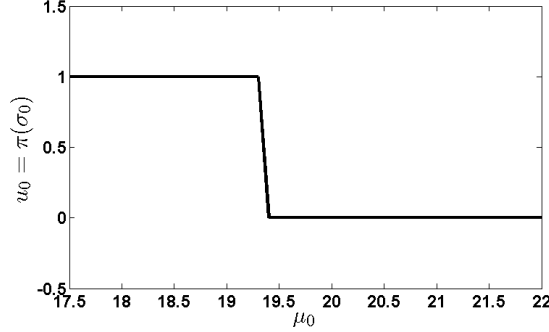


Fig. 2: Given $\sigma_0(x_0, q_0) = \mathbf{1}_{\{0\}}(q_0)\phi(x_0; \mu_0, 0.1)$, optimal choice of u_0 for varying μ_0 . For μ_0 less than or equal to 19.3, it is best to set $u_0 = 1$ (turn the heater on) and after 19.3, the heater should be left off.

continuous state systems to the reachability problem for PODTSHS. We then demonstrated our method on a one dimensional temperature regulation problem with stochastic hybrid dynamics and a noisy discretized measurement of the continuous state. Although the calculation of the α -functions was slow, the policy they encode can be applied quickly online to optimize the system's probability of remaining within a safe region. However, we hope to find more efficient techniques to overcome some of the current method's shortcomings. Over larger state spaces, discretizing the observation space is not practical, and techniques that accomodate a continuous observation space should be explored. The use of particle filters to represent the beliefs may also be beneficial, because of the inability of a small number of Gaussian components to adequately represent discontinuous functions (such as the indicator function). Overall, we believe our method is well-suited to low dimensional systems, and with further investigation should be extendable to higher dimensional systems as well.

REFERENCES

- [1] C. Tomlin, I. Mitchell, A. Bayen, and M. Oishi, "Computational techniques for the verification and control of hybrid systems," in *Proceedings of the IEEE*, vol. 91, no. 7, July 2003, pp. 986–1001.
- [2] M. Prandini and J. Hu, *Stochastic Reachability: Theoretical Foundations and Numerical Approximation*, ser. Lecture Notes in Control and Information Sciences. Springer Verlag, 2006, pp. 107–139.
- [3] I. Mitchell and J. Templeton, "A toolbox of hamilton-jacobi solvers for analysis of nondeterministic continuous and hybrid systems," in *Hybrid Systems: Computation and Control*, 2005, vol. 3414, pp. 480–494.

- [4] A. Abate, M. Prandini, J. Lygeros, and S. Sastry, “Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems,” *Automatica*, vol. 44, no. 11, pp. 2724–2734, 2008.
- [5] S. Summers and J. Lygeros, “Verification of discrete time stochastic hybrid systems: A stochastic reach-avoid decision problem,” *Automatica*, vol. 46, no. 12, pp. 1951–1961, 2010.
- [6] R. Verma and D. del Vecchio, “Control of hybrid automata with hidden modes: Translation to a perfect state information problem,” in *IEEE Conference on Decision and Control*, 2010.
- [7] R. Ghaemi and D. D. Vecchio, “Control for safety specifications of systems with imperfect information on a partial order,” *IEEE Transactions on Automatic Control*, 2014, preprint available online.
- [8] J. Ding, A. Abate, and C. Tomlin, “Optimal control of partially observable discrete time stochastic hybrid systems for safety specifications,” in *American Control Conference*, 2013, pp. 6231–6236.
- [9] K. Lesser and M. Oishi, “Reachability for partially observable discrete time stochastic hybrid systems,” *Automatica*, 2013, submitted, under review.
- [10] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Athena Scientific, 2005, vol. 1.
- [11] S. Soudjani and A. Abate, “Adaptive and sequential gridding procedures for the abstraction and verification of stochastic processes,” *SIAM Journal on Applied Dynamical Systems*, vol. 12, no. 2, pp. 921–956, 2013.
- [12] N. Kariotoglou, S. Summers, T. Summers, M. Kamgarpour, and J. Lygeros, “Approximate dynamic programming for stochastic reachability,” in *European Control Conference*, 2013, pp. 584 – 589.
- [13] C. Lusena, J. Goldsmith, and M. Mundhenk, “Nonapproximability results for partially observable Markov decision processes,” *Journal of Artificial Intelligence Research*, vol. 14, pp. 83–103, 2001.
- [14] G. Shani, J. Pineau, and R. Kaplow, “A survey of point-based POMDP solvers,” *Autonomous Agents and Multi-Agent Systems*, vol. 27, no. 1, pp. 1–51, 2013.
- [15] A. Brooks, A. Makarenko, S. Williams, and H. Durrant-Whyte, “Parametric POMDPs for planning in continuous state spaces,” *Robotics and Autonomous Systems*, vol. 54, no. 11, pp. 887–897, 2006.
- [16] E. Zhou, M. Fu, and S. Marcus, “Solving continuous-state POMDPs via density projection,” *IEEE Transactions on Automatic Control*, vol. 55, no. 5, pp. 1101–1116, 2010.
- [17] J. van den Berg, S. Patil, and R. Alterovitz, “Motion planning under uncertainty using iterative local optimization in belief space,” *The International Journal of Robotics Research*, vol. 31, no. 11, pp. 1263–1278, 2012.
- [18] R. Platt, R. Tedrake, L. Kaelbling, and T. Lozano-Perez, “Belief space planning assuming maximum likelihood observations,” in *Robotics: Science and Systems*, 2010.
- [19] T. Erez and W. Smart, “A scalable method for solving high-dimensional continuous POMDPs using local approximation,” in *26th conference on uncertainty in artificial intelligence*, 2010.
- [20] R. Platt, R. Tedrake, L. Kaelbling, and T. Lozano-Perez, “Efficient planning in non-Gaussian belief spaces and its application to robot grasping,” in *15th International Symposium on Robotics Research*, 2011.
- [21] S. Thrun, “Monte carlo POMDPs,” in *Advances in Neural Information Processing Systems 12*, 2000, pp. 1064–1070.
- [22] J. M. Porta, N. Vlassis, M. T. Spain, and P. Poupart, “Point-based value iteration for continuous POMDPs,” *Journal of Machine Learning Research*, vol. 7, pp. 2329–2367, 2006.
- [23] E. Sondik, “The optimal control of partially observable Markov processes,” Ph.D. dissertation, Stanford University, 1971.
- [24] E. Brunskill, L. Kaelbling, T. Lozano-Perez, and N. Roy, “Planning in partially-observable switching-mode continuous domains,” *Annals of Mathematics and Artificial Intelligence*, vol. 58, pp. 185–216, 2010.

- [25] J. Pineau, G. Gordon, and S. Thrun, “Anytime point-based approximations for large POMDPs,” *Journal of Artificial Intelligence Research*, vol. 27, pp. 335–380, 2006.
- [26] E. Stein and R. Shakarchi, *Real Analysis: Measure Theory, Integration, and Hilbert Spaces*, ser. Princeton Lectures in Analysis. Princeton University Press, 2005.
- [27] I. Tkachev, J.-P. Katoen, A. Mereacre, and A. Abate, “Quantitative automata-based controller synthesis for non-autonomous stochastic hybrid systems,” in *Hybrid Systems: Computation and Control*, 2013, pp. 293–302.
- [28] A. Abate, S. Amin, M. Prandini, J. Lygeros, and S. Sastry, “Computational approaches to reachability analysis of stochastic hybrid systems,” in *Hybrid Systems: Computation and Control*, 2007, vol. 4416.
- [29] K. Zhang and J. Kwok, “Simplifying mixture models through function approximation,” *IEEE Transactions on Neural Networks*, vol. 21, no. 4, pp. 644–658, 2010.